

# BEST AVAILABLE COPY

## PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2003-110604

(43)Date of publication of application : 11.04.2003

(51)Int.Cl.

H04L 12/56  
G06F 13/00  
H04L 29/00  
H04L 29/08

(21)Application number : 2001-306321

(71)Applicant : NIPPON TELEGR & TELEPH CORP  
<NTT>

(22)Date of filing : 02.10.2001

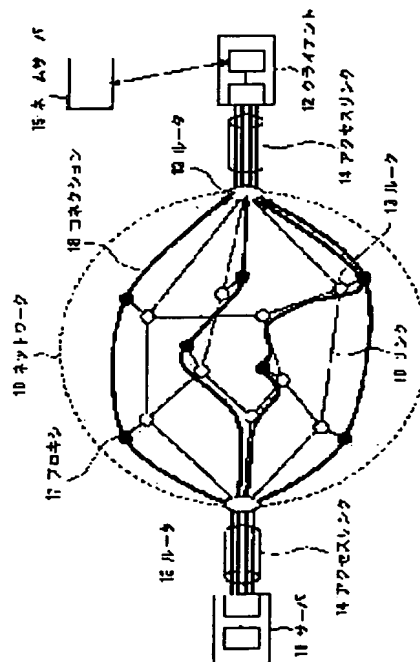
(72)Inventor : ONODA TETSUYA  
FUJII TETSUO

### (54) CLIENT SERVER SYSTEM AND DATA COMMUNICATION METHOD IN THE CLIENT SERVER SYSTEM

#### (57)Abstract:

**PROBLEM TO BE SOLVED:** To provide a client server system with a high-speed access link the throughput of which is enhanced where decrease of the throughput due to congestion of a network or occurrence of a fault is suppressed.

**SOLUTION:** Connections 18 of a plurality of different paths are established between a server 12 and a client 11 and they make data communication through the connections 18. Further, a proxy 17 relays transfer of data between the server 12 and the client 11 by each of the connections 18 in data communication between the server 12 and the client 11.



### LEGAL STATUS

[Date of request for examination]

18.02.2004

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's  
decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開2003-110604

(P2003-110604A)

(43) 公開日 平成15年4月11日 (2003.4.11)

(51) Int.Cl. <sup>7</sup>	識別記号	F I	特コード <sup>*</sup> (参考)
H 0 4 L 12/56	1 0 0	H 0 4 L 12/56	1 0 0 A 5 B 0 8 9
	2 0 0		2 0 0 Z 5 K 0 3 0
G 0 6 F 13/00	3 5 3	G 0 6 F 13/00	3 5 3 C 5 K 0 3 4
H 0 4 L 29/00		H 0 4 L 13/00	3 0 7 A
29/08			S
審査請求 未請求 請求項の数14 O L (全 9 頁)			

(21) 出願番号 特願2001-306321(P2001-306321)

(22) 出願日 平成13年10月2日 (2001.10.2)

(71) 出願人 000004226

日本電信電話株式会社

東京都千代田区大手町二丁目3番1号

(72) 発明者 小野田 哲也

東京都千代田区大手町二丁目3番1号 日

本電信電話株式会社内

(72) 発明者 藤井 哲郎

東京都千代田区大手町二丁目3番1号 日

本電信電話株式会社内

(74) 代理人 100069981

弁理士 吉田 精孝

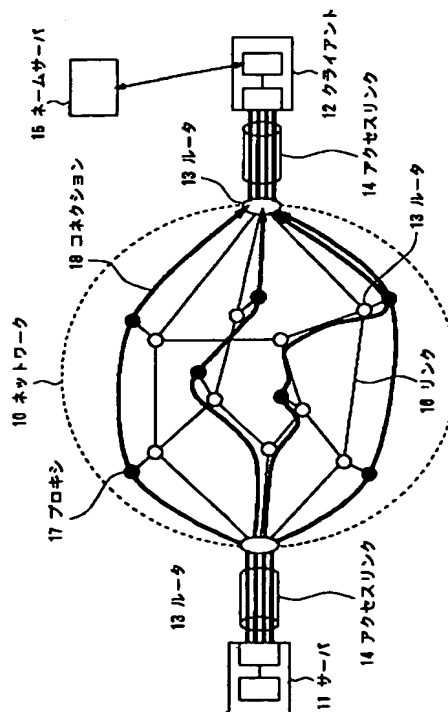
最終頁に続く

(54) 【発明の名称】 クライアントサーバシステム及びクライアントサーバシステムにおけるデータ通信方法

(57) 【要約】

【課題】 高速アクセスリンクを持つクライアントサーバシステムのスループットを向上させ、またネットワークの混雑時や障害時のスループット低下を改善するクライアントサーバシステムを提供する。

【解決手段】 サーバ12・クライアント11間に複数の異なる経路のコネクション18を確立し、該コネクション18上でデータ通信を行う。また、サーバ12・クライアント11間のデータ通信では、コネクション18毎にプロキシ17によりサーバ12・クライアント11間のデータ転送を中継する。



## 【特許請求の範囲】

【請求項 1】 ネットワーク上にクライアント・サーバ間のコネクションを確立し、このコネクションを用いてクライアント・サーバ間でデータ通信を行うクライアントサーバシステムにおいて、ネットワーク上に互いに異なる経路を有する複数のコネクションを確立する手段と、送信データを前記複数のコネクションに分散させて送信するとともに複数のコネクションを介して受信した分散データから元のデータを復元する分散化手段とを備えた、

ことを特徴とするクライアントサーバシステム。

【請求項 2】 前記コネクション確立手段は、前記複数のコネクションのうち少なくとも 1 つ以上のコネクションをプロキシを介して確立することを特徴とする請求項 1 記載のクライアントサーバシステム。

【請求項 3】 コネクション確立手段に対してプロキシのアドレスを提供する手段を備えたことを特徴とする請求項 2 記載のクライアントサーバシステム。

【請求項 4】 前記コネクション確立手段により確立された複数のコネクションのうち少なくとも 1 つ以上のコネクションは複数のプロキシを介することを特徴とする請求項 2 又は 3 何れか 1 項記載のクライアントサーバシステム。

【請求項 5】 各コネクションのスループットを観測する手段を備えるとともに、前記コネクション確立手段は前記スループット観測手段で観測された各スループットの合計が所定の要求条件を満たさない場合に新たなコネクションを追加することを特徴とする請求項 1 乃至 4 何れか 1 項記載のクライアントサーバシステム。

【請求項 6】 前記分散化手段は、データを誤り訂正符号化する手段と、誤り訂正符号化されたデータを復号化する手段とを備え、送信データを符号化手段で誤り訂正符号化し、この誤り訂正符号化された送信データを複数のコネクションに分散して送信するとともに、復元されたデータを復号化手段で復号化することを特徴とする請求項 1 乃至 5 何れか 1 項記載のクライアントサーバシステム。

【請求項 7】 前記ネットワークは TCP/IP プロトコルスイートを用いており、前記コネクションは TCP 層に形成されていることを特徴とする請求項 1 乃至 6 何れか 1 項記載のクライアントサーバシステム。

【請求項 8】 ネットワーク上にクライアント・サーバ間のコネクションを確立し、このコネクションを用いてクライアント・サーバ間でデータ通信を行うクライアントサーバシステムにおいて、データ通信に先立ちネットワーク上に互いに異なる経路を有する複数のコネクションを確立し、データ通信時には、送信データを前記複数のコネクションに分散させて送信するとともに複数のコネクションを

介して受信した分散データから元のデータを復元することを特徴とするクライアントサーバシステムにおけるデータ通信方法。

【請求項 9】 前記コネクションの確立時には、前記複数のコネクションのうち少なくとも 1 つ以上のコネクションをプロキシを介して確立することを特徴とする請求項 8 記載のクライアントサーバシステムにおけるデータ通信方法。

【請求項 10】 前記コネクションの確立時には、アドレス提供手段からプロキシのアドレスを取得し、取得したアドレスを用いてコネクションを確立することを特徴とする請求項 9 記載のクライアントサーバシステム。

【請求項 11】 前記コネクションの確立時には、少なくとも 1 つ以上のコネクションが複数のプロキシを介するようにコネクションを確立することを特徴とする請求項 9 又は 10 何れか 1 項記載のクライアントサーバシステムにおけるデータ通信方法。

【請求項 12】 各コネクションのスループットを観測し、スループット観測手段で観測された各スループットの合計が所定の要求条件を満たさない場合に新たなコネクションを追加することを特徴とする請求項 8 乃至 11 何れか 1 項記載のクライアントサーバシステムにおけるデータ通信方法。

【請求項 13】 送信データを符号化手段で誤り訂正符号化し、この誤り訂正符号化された送信データを複数のコネクションに分散して送信するとともに、復元されたデータを復号化手段で復号化することを特徴とする請求項 8 乃至 12 何れか 1 項記載のクライアントサーバシステムにおけるデータ通信方法。

【請求項 14】 前記ネットワークは TCP/IP プロトコルスイートを用いており、前記コネクションは TCP 層に形成されていることを特徴とする請求項 8 乃至 13 何れか 1 項記載のクライアントサーバシステム。

## 【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、クライアントサーバシステムに関するものである。

【0002】

【従来の技術】インターネットに代表される TCP/IP プロトコルスイートを用いたネットワーク上のクライアントサーバシステムでは、サーバのアドレス、ポート番号とクライアントのアドレス、ポート番号との間でコネクションの確立が行われた後、これらの間のデータのやりとりは IP データグラムというパケットに分割されて、インターネット上を転送させる。インターネットではルータが IP データグラムに付与された宛先 IP アドレスを確認し、それぞれルータが持つ経路テーブルに基づいてパケットが転送され、サーバ、クライアント間のデータのやりとりが行われる。現状、ルータは経路テーブルを OSPF (Open Shortest Pat

h Fast)等のルーティングプロトコルを用いてメンテナンスしている。OSPFなどのルーティングプロトコルでは自分に隣接するルータまでのコスト(通常はインタフェース速度の逆数)をLSA(Link State Advertisement)として自分に隣接するルータに広告通知する。この広告通知されたLSAにより各ルータは経路テーブルを構築する。この経路テーブルを用いることで宛先IPアドレスに対し、最小コストの経路が選択され、ルータから転送される。したがって障害などによる経路情報の変化がなければ、任意のIPアドレス間のデータ転送経路は一意に定まり、常にこの経路をパケットが通過することになる。

【0003】従来のクライアントサーバシステムにおけるファイルデータ転送について、以下に説明する。まず、従来のルータ装置の構成について図6を参照して説明する。

【0004】図6は、従来のクライアントサーバシステムにおけるファイルデータ転送の例を示す図である。符号1はクライアント、2はサーバ、3はルータ、4はアクセスリンク、5はネームサーバ、6はインターネット上のリンクである。

【0005】例として、クライアント1がftp(File Transfer Protocol)を用いてサーバ2からファイルを転送する場合を取り上げる。クライアント1は、まず、ftpのためのTCPコネクションをサーバ2との間で確立する。次に、クライアント1は、ネームサーバ5へサーバ2のアドレスを問い合わせる。そして、ネームサーバ5より回答されたサーバ2のIPアドレスに対し、クライアント1はftpのためのTCPのコネクションを確立し、ftpのコマンド等をサーバ2との間でやりとりする。確立されたTCPコネクション上のデータは、IPデータグラム(パケット)に(必要があれば分割されて)格納され、アクセスリンク4、ルータ3、ルータ間を結ぶリンク6を介してクライアントサーバ間をやりとりされる。この時、サーバ、クライアント間のパケットが流れる経路は各ルータによって選択されるがOSPFなどのルーティングプロトコルは最小コスト経路を常に選択するため、障害などでリンクが不通となっている場合などを除いて常に同一経路が選択されて転送される。

【0006】

【発明が解決しようとする課題】上記のように従来のクライアントサーバシステムでは、データのやりとりにインターネット上の一つの経路が利用される。画像などのマルチメディアアプリケーションや大きなファイルの転送のように高速でのデータのやり取りが必要なユーザはADSLやケーブルモデム、FTTHといった高速のアクセスリンクを契約し、クライアントサーバ間で高速データ転送を試みる。しかし、アクセスリンクが高速であってもインターネット上の経路は一定であり、他のパケ

ットとそれぞれのリンクでの使用可能帯域をシェアすることになる。このため、サーバ・クライアント間のスループットは経路上のリンクの最小帯域に制限される。また、経路上の何れかのリンクが混雑していれば、他のリンクが空いていたとしても迂回することができず、他のパケットと同様にスループットの低下を招いてしまう。このように従来のクライアントサーバシステムではサーバ・クライアント間に最小コスト経路が設定されてしまうため、アクセスリンクに高速リンクを用意したとしてもサーバクライアント間に高スループットを期待することができず、また混雑時の迂回や障害を避けるために最小コスト以外の経路による冗長な転送が不可能であった。

【0007】本発明は、上記事情に鑑みてなされたものであり、その目的とするところは、高速アクセスリンクを持つクライアントサーバシステムのスループットを向上させ、またネットワークの混雑時や障害時のスループット低下を改善するクライアントサーバシステムを提供することを目的とする。

【0008】

【課題を解決するための手段】上記目的を達成するために、本発明では、サーバ・クライアント間に複数の異なる経路のコネクションを確立し、該コネクション上でデータ通信を行うものを提案する。また、サーバ・クライアント間のデータ通信は、両者間で直接データをやり取りするのではなく、コネクション毎にプロキシによりサーバ・クライアント間のデータ転送を中継する方法を提案する。本発明によれば、一つのアプリケーションに対し、複数のコネクションを同時に利用し、更にそれぞれの経路を変更することでサーバ・クライアント間のスループット向上を図り、混雑や障害によるスループット低下を改善する。

【0009】さらに、本発明では、中継するプロキシを一段ではなく、多段接続するものを提案する。これは、一端経路が分散された複数のコネクションが同一リンク上を通過することを防ぐことを目的としており、経路分散を確実にすることでスループット向上を図る。

【0010】さらに、本発明では、ネームサーバの変わりに負荷分散装置を用いて各コネクションを中継するプロキシを選択するものを提案する。ネームサーバに特定サーバのアドレスとして複数のプロキシのアドレスを登録しておくのではなく、通常どおりのサーバのアドレスを格納しておく。サーバ・クライアント間の経路上、例えばクライアントの隣接する位置に負荷分散装置を設置し、同一のサーバへの複数コネクション(クライアントのポート番号が異なるため識別可能)によるアクセスをそれぞれ異なるプロキシへ振ることにより経路を分散させる。

【0011】さらに、本発明では、サーバ・クライアント間の経路分散化されたコネクションをデータ転送に信

頼性向上の目的で、データを誤り訂正符号化する方法を提案する。サーバ側で誤り訂正符号化し、クライアント側で復号化することにより、一部の経路上に障害が生じた場合でも、もとのデータを復元することができる。

【0012】

【発明の実施の形態】（第1の実施の形態）本発明の第1の実施形態にかかるクライアントサーバシステムについて図面を参照して詳細に説明する。

【0013】図1は本発明の実施形態にかかるネットワークの構成図である。図中、10はネットワーク、11はクライアント、12はサーバ、13はルータ、14はアクセスリンク、15はネームサーバ、16はネットワーク上のリンク、17はプロキシ、18はコネクションである。ここで、ネットワーク10としては、例えばインターネットなどのTCP/IPプロトコルスイートを用いたネットワークを用いた。また、アクセスリンク14、リンク16はコネクション18の下位の層である。本実施の形態ではネットワーク10としてTCP/IPを用いているので、IP層以下に相当するものでありOS参照モデルのネットワーク層以下にほぼ相当する。また、本実施の形態では、コネクション18はTCP/UDP層でありOS参照モデルのトランスポート層にほぼ相当する。

【0014】本実施の形態ではサーバからクライアントへのファイル転送を例として説明する。また、ここでは、並列経路を使ってファイルを転送するアプリケーションを想定し、このアプリケーションのためのサーバ側のポート番号xがクライアント側で既知であるとする。

【0015】クライアント11は、ftpと同様にサーバ12のポート番号xへのコネクション18を確立するため、ネームサーバ15へサーバ12のIPアドレスを問い合わせる。この時、ネームサーバ15には、実際のサーバ12のIPアドレスではなく、サーバ・クライアント間のデータ転送を中継するプロキシ17のアドレスが格納されており、ネームサーバ15はこれをサーバ12のIPアドレスへの問い合わせに対する回答として答える。クライアント11は、サーバのポート番号xに対して複数のTCPコネクション18を確立しようとし、そのたびにネームサーバ15へIPアドレスを問い合わせる。

【0016】ネームサーバ15は複数登録されているプロキシ17のアドレスを例えばラウンドロビンなどのルールで順次クライアント11へ回答する。具体的にはネームサーバ15にはサーバ12のアドレス（例えばsss.sss.sss.sss）の代わりに、以下のようにプロキシサーバのIPアドレスがAレコードとして加えられている。

servername A xxx.xxx.xxx.xxx(第1のプロキシのIPアドレス)

servername A yyy.yyy.yyy.yyy(第2のプロキシのIPアドレス)

servername A zzz.zzz.zzz.zzz(第3のプロキシのIPアドレス)

これにより、ネームサーバ15はクライアント11からのIPアドレスの問い合わせに対して、サーバ12のIPアドレスではなく、プロキシ15のIPアドレスを問い合わせ順にラウンドロビンで回答する。

【0017】ネームサーバ15からIPアドレスの返答を受けたクライアント11はサーバ12のTCPコネクション18の確立要求"SYN(Synchronizeの意)"をコネクション18ごとに異なるプロキシ17のポート番号xへ送信する。

【0018】各プロキシ17は、ポート番号xへのアクセスを並列転送であると認識し、サーバ12のIPアドレス、ポート番号xへ中継する。すなわち、プロキシ17には、クライアント11からのアクセスをサーバ12へ中継するために予めTCPデリゲイトが設定されている。例えば、「delegated」というプロキシプログラムを用いる場合、

delegated -P5000 SERVER=tcprelay://sss.sss.sss.sss:6000

という設定がなされている。ここで、sss.sss.sss.sssはサーバ12のIPアドレス、6000はサーバ12における並列転送のためのポート番号であるとする。また、5000はプロキシ17のポート番号xで、クライアント11から5000番のTCPポートにアクセスされた場合、クライアント11からのアクセスをサーバ12に中継するという意味である。プロキシ17は、クライアント11からアクセスされたポート番号によって中継する先のサーバ12を識別することができる。

【0019】なお、クライアント11とサーバ12間のコネクション18においてプロキシ17による中継点は2以上あってもよい。すなわち、図1に示すように、一部のプロキシ17のポート番号xへのアクセスをサーバ12へ中継するのではなく、多段接続のため、他のプロキシ17へ中継してもよい。この場合には、多段接続の最終プロキシ17が最終的にサーバ12のIPアドレス、ポート番号xへコネクション確立要求を転送する。

【0020】プロキシ17を介してクライアント11から複数のコネクション確立要求"SYN"を受けたサーバ12は、ポートxへのアクセスに対して、これらを並列転送のアプリケーションと認識してそれぞれTCPコネクション18を確立する。これらは並列転送のためのコネクション18であるため、同時に複数のTCPコネクション18の利用を想定して、同一ポートxに対する複数のコネクション要求を受け付ける。プロキシ17を経由しているため、この時バインドされるクライアント側のアドレスはプロキシ17のIPアドレスとなる（プロキシ17が多段接続されている場合は最終段のプロキシ17のIPアドレス）。TCPコネクション18の確立要求に対して、サーバ12は、"ACK(Ackno

wledgeの意)”と同時にサーバ12からクライアント11側への逆方向のTCPコネクション要求”SYN”を送信する。サーバ12からそれぞれのプロキシ17に対して送信されたACK、SYNはクライアント11からサーバ12への転送中継経路の逆を辿ってクライアント11へ到着する。

【0021】プロキシ17を介してサーバ12からのACK、SYNを受け取ったクライアント11は、ACKによりプロキシ17への送信TCPコネクション18が確立されたことを知り、またSYNにより、サーバ12からクライアント11への逆方向へのTCPコネクション18の確立要求に対してACKを返す。これにより、サーバ12・クライアント11間にプロキシ17を介した複数の双方向TCPコネクション18が確立されたことにある。以上、TCPサーバ・クライアント間のコネクション確立にかかるハンドシェイク図を図2に示す。また、図3に本発明による並列転送のプロトコルスタックを示す。図3に示すように、ユーザプログラムがTCPコネクション18を利用する際のインタフェースを”ソケット”と呼ぶ。

【0022】コネクション18が確立された後、クライアント11は確立されたTCPコネクション18の何れかを用いてアクセスするファイル名をサーバ12に送信する。ファイル名の転送は、クライアント11側ではTCPソケットへのライトであり、サーバ12側ではTCPソケットからのリードである。クライアント11は、コネクション18が確立された後、ライト可能な、即ちソケットバッファに空きが存在するTCPソケットの何れかを選択し、ファイル名をライトする。サーバ12側では、リード可能な、即ちソケットバッファが空き状態ではないTCPソケットを選択し、リードする。クライアント11側、サーバ12側のプログラムはそれぞれTCPソケットの状態（バッファの空きの有無など）をポーリングしており、ライト可能、リード可能なTCPソケットを選択するようにする。TCPコネクション18によるデータの転送中、これを中継するプロキシ17では転送にかかるIPパケットの送り元アドレスをプロキシ17自身へ、宛先アドレスをプロキシ17自身のアドレスから中継経路上の次のアドレス（サーバ12、クライアント11、あるいは次の中継プロキシ17のアドレス）へ変更する。

【0023】TCPコネクション18の何れかによりファイル名を受信したサーバ12は、自身に格納するファイルにアクセスし、ファイルデータをTCPコネクション18を通じてクライアント11へ送信する。サーバ12のプログラム（以下サーバプログラムと言う。）は、クライアント11側でファイルデータを再構築できるように、ファイルデータをTCPコネクション18に送信する際（TCPソケットへライトする際）、データのファイル上における位置をシーケンス番号としてライトす

る。サーバプログラムは、ライト可能な（すなわち、ソケットバッファに空きがある）TCPソケットをポーリングにより識別し、順にファイルデータをシーケンス番号を付与しながらライトしていく。プロキシ17により中継されたファイルデータがクライアント11側のTCPソケットに到着するとソケットバッファのデータ量が空き状態から変化する。クライアント11側のプログラムはポーリングにより、ソケットバッファへのデータの到着を認識し、空きでないソケットバッファから順次ファイルデータを読み出す。データにはシーケンス番号が付与されているため、複数のTCPコネクション18による並列転送でファイルデータの到着順序が入れ替わった場合でも、クライアント11側のプログラムはファイルデータの再構築ができる。

【0024】本実施形態ではサーバ12・クライアント11間のコネクション18にTCPを用いているため、ネットワーク上でIPデータグラムのパケットロスやエラーが発生した場合、再送要求・再送を行い、確実にデータを転送する。転送の発生や、輻輳による遅延などにより、TCPのスループットは、経路により異なる。サーバ側のプログラムはバッファに空きが発生したTCPソケットをポーリングにより検知し、順次ファイルデータをライトする。パケットの到着を確認するACKがクライアントから返されなければ、再送の可能性があるためバッファはデータを捨てることができず、バッファは空き状態とならない。すなわち、ライト可能なTCPソケットを順次利用することでそれぞれTCPコネクション18のスループットに応じてサーバプログラムはファイルデータの転送量を変化させることができる。クライアント11側では各TCPコネクション18毎のスループットを観測できる。そこで、各TCPコネクション18のスループットを観測し、更にその合計値が要求条件（動画ファイルダウンロードしながら再生する疑似ストリーミングの再生速度等）を満たさない場合、クライアント11側からの上述の手順により更にTCPコネクション18の確立要求をサーバ12側に送信し、あらたにコネクション18を追加することにより、全体のスループットを改善するという方法も取りうる。

【0025】サーバプログラムは全てのファイルデータの転送が終了した後、ライト可能となったTCPソケットに対し、EOF（End of File）を転送する。クライアント11側でEOFを受信したTCPソケットはファイルデータの終了と認識し、TCPコネクション18をクローズする。全てのコネクション18がクローズされた場合、ファイルデータは全て受信されたこととなり、クライアント11側のファイル転送プログラムは終了する。

【0026】次に、クライアント・サーバ間の通信データを複数のコネクション18に分散して送受信する際の、元の通信データと各コネクション18を通るデータ

との関係について説明する。

【0027】本実施の形態では、データを誤り訂正符号化により冗長な情報に符号化し、これを複数の経路に分散して転送することにより、経路上のパケットロスやエラーが存在しても、もとのデータを復元できるというものである。例えばサーバ12では誤り訂正符号としてリードソロモン符号を用いて各々のファイルデータのバイト列情報を符号化し、データDとチェックサムCを生成する（チェックサムはVandermonde行列Fにより算出するものとする）。このとき、上述の例とは異なり、ソケットにTCPではなくUDPを使用することもある。UDPはTCPと違い、再送メカニズムを持たず、信頼性のある転送を行うことができないが、フローコントロール等の複雑な転送メカニズムを持たないため、簡便で高速な転送が可能である（フローコントロールを行わないため、それぞれのコネクション18の転送速度については上限を設けるなどして他のTCPフローへの影響を考慮する必要がある）。クライアント11では各コネクション18から得られた情報をもとに復号化を行い、もとの情報を復号する。復号化に使う行列

$$I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad F = \begin{bmatrix} 1^0 & 2^0 & 3^0 \\ 1^1 & 2^1 & 3^1 \\ 1^2 & 2^2 & 3^2 \end{bmatrix}$$

【0032】である。

【0033】いま例として、 $d_1=3$ 、 $d_2=1$ 、 $d_3=9$ というデータ列を考える（図4参照）。上の式でチェックサムを計算すると、 $c_1=11$ 、 $c_2=9$ 、 $c_3=12$ となる。すなわち、

【0034】

【数3】

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 2 & 3 \\ 1 & 4 & 9 \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \\ d_3 \end{bmatrix} = \begin{bmatrix} 3 \\ 1 \\ 9 \\ 13 \\ 32 \\ 88 \end{bmatrix}$$

【0035】これら合計6つのデータを別々の経路を通じてクライアント11に送信する。送信途中で仮に3つの経路に障害があり、 $d_2$ 、 $d_3$ 、 $c_3$ のデータが失われたと仮定する。しかし、クライアント11側では正常に受信され他3つデータからこれらを復元することができる。具体的には、復号化では届かなかった $d_2$ 、 $d_3$ 、 $c_3$ に対応する行を上式の式から消去すればよく、

【0036】

【数4】

はサーバ12側と同様にVandermonde行列である。

【0028】符号化・復号化の例を以下に示す。ここではデータ（バイト列）を表現する行列をD、チェックサムを表現する行列をCとして表現する（例では3バイトの情報 $\{d_1, d_2, d_3\}=D$ 、これに対するチェックサム $\{c_1, c_2, c_3\}=C$ として表現する。したがって転送経路は $\{d_1, d_2, d_3, c_1, c_2, c_3\}$ を転送するための6経路とする。Vandermonde行列をF、対角行列をIと表現する。符号化・復号化は、それぞれ以下のように表される。この時、Fの各要素はガロア体の乗算で求められる。

【0029】

【数1】

$$\begin{bmatrix} I \\ F \end{bmatrix} D = C$$

【0030】Dを3行の行列とすると、

【0031】

20 【数2】

$$\begin{bmatrix} 1^0 & 2^0 & 3^0 \\ 1^1 & 2^1 & 3^1 \\ 1^2 & 2^2 & 3^2 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 3 \\ 1 & 4 & 9 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 2 & 3 \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \\ d_3 \end{bmatrix} = \begin{bmatrix} 3 \\ 13 \\ 32 \end{bmatrix}$$

30

【0037】となる。これからガウスの消去法で逆行列を計算すると、

【0038】

【数5】

$$\begin{bmatrix} d_1 \\ d_2 \\ d_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ -2 & 3 & -1 \\ 1 & -2 & 1 \end{bmatrix} \begin{bmatrix} 3 \\ 13 \\ 32 \end{bmatrix}$$

40

【0039】となり、もとのデータ $d_1=3$ 、 $d_2=1$ 、 $d_3=9$ をクライアント12側で復元することができるというものである。これにより、例えば信頼性の低いUDPコネクションを用いたとしても、誤り訂正符号を用いて経路を分散して送信することで、信頼性を向上させることができる。

【0040】（第2の実施の形態）本発明の第2の実施形態にかかるクライアントサーバシステムについて図5を参照して詳細に説明する。図5は第2の実施の形態にかかるクライアントサーバシステムの全体構成図であ

50 る。なお、図中、第1の実施の形態と同様の構成には同



一の符号を付した。

【0041】本実施の形態にかかるクライアントサーバシステムが第1の実施の形態と異なる点は、プロキシ17のアドレスを取得するために、ネームサーバの代わりに負荷分散装置21を用いる点にある。ネームサーバ25は通常どおり、サーバ12の実際のアドレスをクライアント11に回答する。クライアント11からの複数のTCPコネクション確立要求は全てサーバ12のアドレスに向けて送信されるが、この経路上（例えば図5のようにクライアント11に隣接の位置）に存在する負荷分散装置21がサーバ12のIPアドレス毎に複数のプロキシ17のIPアドレスを登録しておき、サーバ12のIPアドレス、ポート番号x向けのTCPコネクション要求を複数の異なるプロキシ17へ経路を分散し、送信する。これにより、ネームサーバ25にプロキシのアドレスを登録することなく、同一サーバ12へのTCPコネクション18の経路を分散させることができる。その他の構成・方法については第1の実施の形態と同様である。

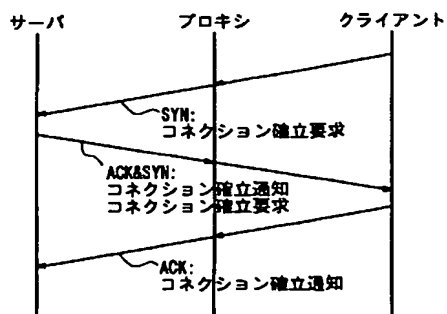
【0042】以上本発明の実施形態について説明したが本発明はこれに限定されるものではない。本発明の範囲は特許請求の範囲によって示されており、全ての変形例は本発明に含まれるものである。

【0043】例えば、上記各実施の形態では、クライアント・サーバ間の全てのコネクションがプロキシを介して形成されているが、コネクションが異なる経路で複数経路で形成できれば、プロキシを介さないコネクションを含んでいてもよい。この場合、プロキシを介さないコネクションについては、従来のルーティングルールにより形成すればよい。

【0044】また、上記第2の実施の形態では、負荷分散装置をクライアントの近傍に設けたが他の場所であってもよい。

【0045】さらに、上記各実施の形態では、ネットワーク10としてインターネットを例示したが他のネット

【図2】



ワークでもあってよい。また、データ通信に用いるプロトコルもTCP/IP以外のプロトコルであってもよい。

【0046】

【発明の効果】以上のように本発明では、サーバ・クライアント間のデータ転送を異なる経路を持つ複数のコネクションにより実現する。このため、1) 高速アクセスリンクを用意した場合でもデータ転送経路が1つであるためアクセスリンク速度に見合った高スループットが得られない、2) 経路上の一部のリンクが輻輳や障害があった場合に直接その影響を受ける、といった問題点を改善することができる。また、経路を分散させることにより、個別のコネクションが低スループットであっても全体で高スループットを実現でき、更に誤り訂正符号を用いるなど冗長な情報を経路を分散させて転送させることで、従来の1ルートのみによる転送に比べて信頼性を向上させることができる。

【図面の簡単な説明】

【図1】第1の実施形態にかかるクライアントサーバシステムの構成図

【図2】サーバ・クライアント間のコネクション確立にかかるハンドシェイク図

【図3】並列転送のプロトコルスタックを説明する図

【図4】複数の経路における障害が発生した場合の処理を説明する図

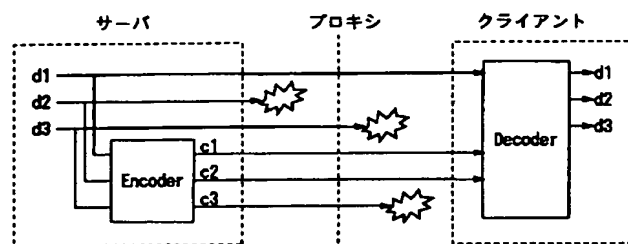
【図5】第2の実施形態にかかるクライアントサーバシステムの構成図

【図6】従来のネットワークにおける通信経路を説明する図

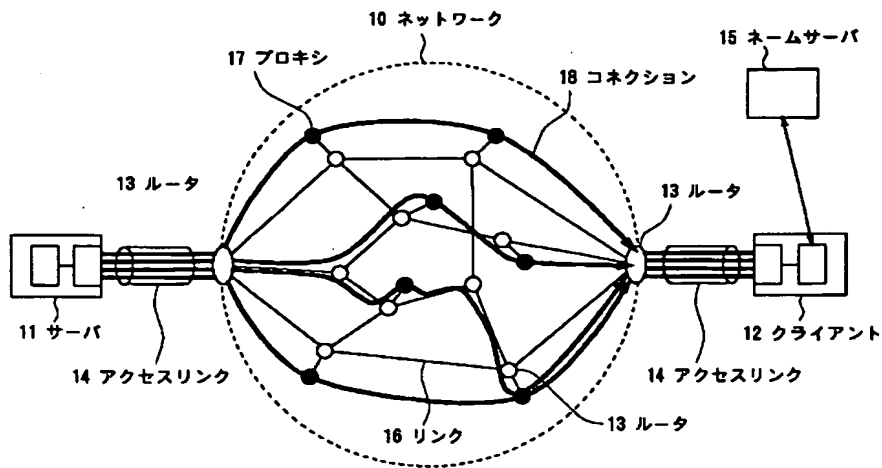
【符号の説明】

10…ネットワーク、11…クライアント、12…サーバ、13…ルータ、14…アクセスリンク、15、25…ネームサーバ、16…リンク、17…プロキシ、18…コネクション、21…負荷分散装置

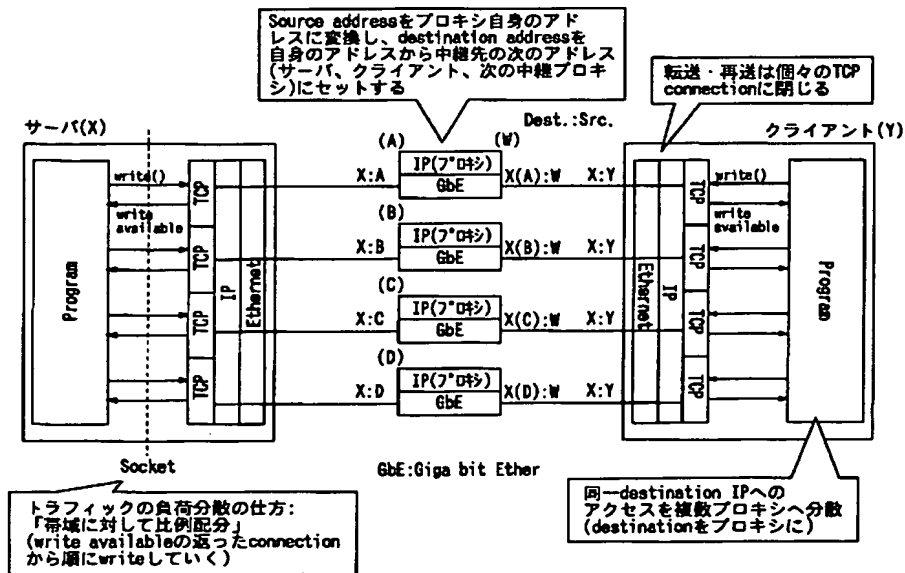
【図4】



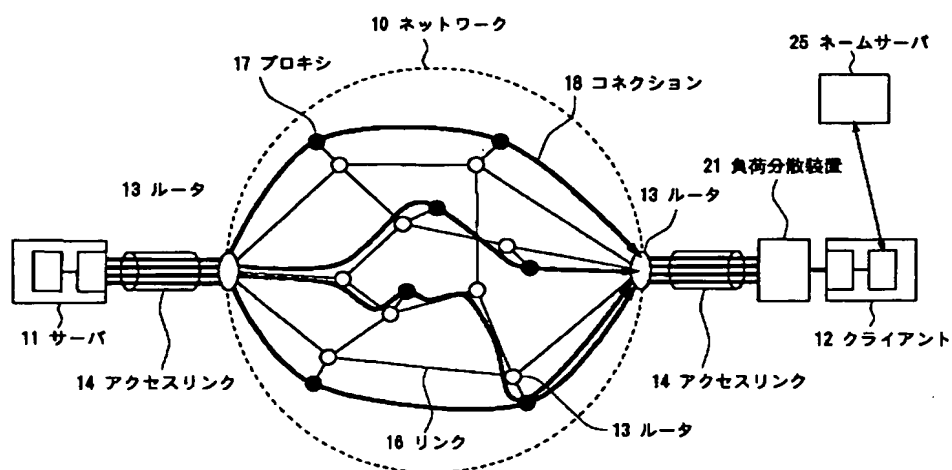
【図1】



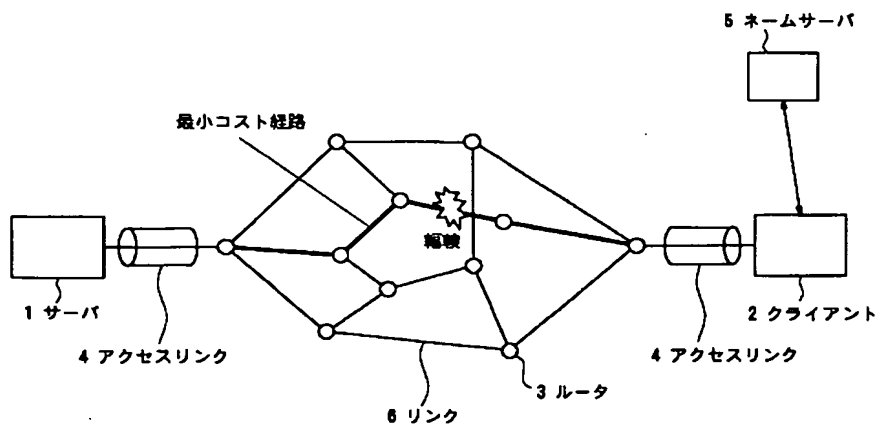
【図3】



【図5】



【図6】



フロントページの続き

Fターム(参考) 5B089 GA11 GA19 GA21 GB09 HB02  
 JA32 KA07 KA12 KG04 MA07  
 5K030 GA03 HA08 JT03 JT06 LC08  
 LC11 LE03 MB03  
 5K034 AA01 AA03 EE07 EE10 FF02  
 HH01 HH02 HH12 HH18 LL01  
 MM25

**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

**BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

☐ BLACK BORDERS

☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES

☒ FADED TEXT OR DRAWING

☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING

☐ SKEWED/SLANTED IMAGES

☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS

☐ GRAY SCALE DOCUMENTS

☒ LINES OR MARKS ON ORIGINAL DOCUMENT

☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY

☐ OTHER: \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**